



PRD — Assistant pédagogique IA complet

Plateforme Symfony + API Platform + IA — Projet MMI3

1. Objectifs du projet

Créer une plateforme permettant à un formateur de :

1. **Importer un syllabus** (PDF ou texte brut)
2. **Générer automatiquement :**
 - un **plan de cours complet**
 - une **progression pédagogique**
 - des **objectifs par séance**
 - des **exercices / activités**
 - des **critères d'évaluation**
3. **Suivre la progression réelle** des étudiants
4. **Adapter les prochaines séances** grâce à un assistant IA contextuel
5. **Disposer d'un chat IA** spécialisé sur :
 - le syllabus
 - la progression réelle
 - le niveau du groupe
 - les compétences visées



Ce que l'enseignant obtient réellement :

Un assistant qui prépare les cours, les adapte après chaque séance, génère des exercices, corrige des productions simples et propose un suivi individuel.

2. Personas

Formateur

- Gère les groupes
- Importe les syllabus
- Valide ou ajuste les plans de cours générés
- Marque une séance comme "faite"
- Consulte les dashboards

Étudiant (facultatif pour MVP)

- Consulte sa progression
- Reçoit des exercices
- Répond (si module évaluation activé)

IA (fournie via API externe)

- Génère du contenu pédagogique structuré
- Analyse le syllabus
- Suit l'évolution du groupe
- Propose des ajustements

3. Fonctionnalités majeures (scope détaillé)

3.1. Import du syllabus

Objectif : fournir une base de travail à l'IA

Formats acceptés

- PDF
- Texte brut (copié/collé)
- Markdown (optionnel)

Fonctionnement

1. Upload du fichier
2. Extraction du texte (automatisée ou manuelle)
3. Stockage en base
4. Passage à l'IA pour analyse

Données extraites

- compétences
- thématiques
- prérequis
- objectifs globaux
- durée approximative

Les étudiants devront décider : extraction automatique OU copier-coller manuel (simpler).

3.2. 🤖 Génération automatique du plan de cours

Endpoint proposé :

POST /ai/generate-course-plan

Entrées JSON :

```
{
  "syllabus": "texte complet",
  "nombre_seances": 8,
  "duree_par_seance": 3,
  "niveau_etudiants": "intermédiaire",
  "competences": ["Symfony", "API Platform"]
}
```

Sortie (JSON structuré) :

```
{
  "plan_general": "...",
  "seances": [
    {
      "titre": "Introduction à Symfony",
      "objectifs": [],

```

```
    "contenus": [],
    "activites": [],
    "ressources": []
  },
],
"evaluation": {
  "modalites": [],
  "criteres": []
}
}
```

MVP obligatoire :

- ✓ Analyse du syllabus
- ✓ Génération du plan de cours
- ✓ Génération des séances
- ✓ Génération des activités
- ✗ Pas besoin d'export PDF
- ✗ Pas besoin de timeline visuelle

3.3. Suivi de progression

Le formateur peut :

- marquer une séance comme "faite"
- indiquer ce qui a été réellement vu
- demander à l'IA une adaptation de la séance suivante

Endpoints

- `PATCH /sessions/{id}/complete`
- `POST /ai/update-course-plan` (réajustement IA)

3.4. Génération d'exercices & corrections (IA)

Génération d'exercices

```
POST /ai/generate-exercises
```

Inputs :

- difficulté
- compétence cible
- contexte séance

Outputs :

- liste d'exercices
- barème simple
- exemples de correction

Correction IA (bonus)

`POST /ai/correct`

Inputs :

- réponse étudiant
- consigne
- niveau attendu

Outputs :

- feedback
 - score (facultatif)
-

3.5. Chat IA contextuel

Un "ChatGPT pédagogique" spécialisé :

- contexte = syllabus + progression + objectifs pédagogiques
 - sert à :
 - reformuler
 - générer des explications adaptées
 - produire des exemples
 - répondre aux questions du formateur
-

4. Modèle de données (proposé)

Syllabus

- id
- title
- raw_text
- extracted_competences (JSON)
- created_at

CoursePlan

- id
- syllabus_id
- plan_general
- evaluation (JSON)
- sessions (relation)

Session

- id
- course_plan_id
- titre
- objectifs (JSON)
- contenus (JSON)
- activites (JSON)
- done (bool)
- notes_reelles (text)

Exercise

- id
 - session_id
 - consigne
 - correction (JSON)
-

5. Architecture technique

Backend

- Symfony 7
- API Platform
- Security : JWT (LexikJWT à leur choix)
- VichUploaderBundle (optional pour PDF)
- Appels IA : HTTP client Symfony

Front (choix étudiants)

- React / Vue / aucune interface (Swagger only)
- Ou Symfony UX base minimale

IA

- API OpenAI / Mistral / LM Studio local
- Format strict JSON demandé pour parse simplifié

6. Sécurité / authentication

MVP recommandé :

- Login formateur
- Rôle unique : ROLE_TEACHER
- Possibilité d'ajouter ROLE_ADMIN plus tard

7. Choix critiques que les étudiants devront trancher

1. Extraction du PDF :

- automatique (lib extra, complexité élevée)

- manuelle (copier-coller → simple)

2. Stockage du plan de cours :

- en JSON dans une colonne unique ?
- en entités reliées (plus propre) ?

3. Appels IA :

- gros prompt → sortie complète
- prompts séparés (compétences, séances, évals)

4. Front ou pas ?

- API only (plus réa tech)
- petit front vite fait (meilleure UX)

5. Sécurité :

- JWT (pro)
- session classique (plus simple)

6. Scope du MVP :

Recommandé :

- import syllabus
- génération plan de cours
- gestion séances
- suivi progression

8. Roadmap proposée (pour 3 semaines)

Semaine 1 — Fondation

- Setup Symfony
- Setup API Platform
- Mise en place entités : Syllabus, CoursePlan, Session
- Endpoint dossier → upload + stockage
- Endpoint IA → génération plan de cours

Semaine 2 — Fonctionnel

- CRUD Sessions
- Marquer une séance comme faite
- Réajustement IA du plan de cours
- Génération d'exercices

Semaine 3 — Intelligence + Bonus

- Chat contextuel
- Dashboard progression
- Amélioration prompts
- Corrections IA (optionnel)

9. ✓ Livrables attendus

Obligatoires

- API REST fonctionnelle
- Documentation Swagger complète
- 1 ou plusieurs endpoints IA
- Modèle de données cohérent
- Code propre, contrôleurs légers
- README expliquant :
 - les choix techniques
 - les limites
 - les prochaines features

Optionnels

- Petit front
- Dashboard

- Chat IA
 - Export PDF
 - Correction automatique
-

10. Critères d'évaluation

Technique

- Modèle de données solide
- Architecture propre
- Cohérence API Platform
- Bonne gestion des erreurs
- Qualité des prompts IA

Pédagogie du produit

- Pertinence du plan généré
- Clarté du syllabus analysé
- Qualité du suivi pédagogique

UX / UI (si front)

- simplicité
- efficacité
- accessibilité

Bonus

- Originalité
- Qualité du chat IA
- Automatisation pertinente